AMENDMENTS TO THE SPECIFICATION

Please amend the title with the following title:

DATA AREA MANAGING METHOD IN INFORMATION RECORDING MEDIUM AND INFORMATION PROCESSOR EMPLOYING DATA AREA MANAGING ~~METHOD~~

Please substitute the paragraphs beginning on page 5, line 23 and ending at page 6, line 5 to read as follows:

~~Fig. 2 is a view~~ Figs. 2A-2C are views showing an example of the data storage in a FAT file system.

~~Fig. 3 is a view~~ Figs. 3A-3D are views showing an example of the writing of file data in the FAT file system.

Fig. 4 is a view showing an example of a file system constructed on the information recording medium.

~~Fig. 5 is a view~~ Figs. 5A-5B are views showing an example of a FAT cache in embodiment 1.

Please substitute the paragraph beginning on page 6, line 13 and ending at line 14 to read as follows:

~~Fig. 9 is a view~~ Figs. 9A-9B are views showing an example of a FAT_Read and FAT_Write cache in embodiment 2.

Please substitute the paragraphs beginning on page 9, line 8 and ending at page 11, line 16 to read as follows:

Referring to Fig. 2, an example of reading the file data in the FAT file system will be described. As shown in Fig. ~~2(A)~~ 2A, the root directory entry 114 and data area 115 store, in a part of them, a directory entry 201 storing a file name, file size, and etc. The data area which stores file data is managed in units of clusters. A uniquely distinguishable cluster number is given to each cluster. To identify the cluster storing the file data therein, a cluster number of the cluster storing a head of the file data, that is, a start cluster number is stored in the directory entry 201. The example of the directory entry 201 in Fig. ~~2(A)~~ 2A shows that a file named as FILE1. TXT stores data starting from the cluster number 10.

Concerning a file whose data is stored in a plurality of clusters, it is necessary to identify the cluster number succeeding the start cluster number and track the clusters which store the data therein. Thus, link information of the clusters

is stored in the FAT. Fig. ~~2(B)~~ 2B shows an example of the FAT 202. FAT 202 is provided with fields corresponding to the respective cluster number. Each field stores a FAT entry indicating the link information of each cluster therein. The FAT entry indicates the cluster number of the cluster to be linked with next. In the example in Fig. ~~2(B)~~ 2B, "11" is stored as the FAT entry corresponding to the cluster number 10. This means that the cluster with the cluster number 10 is linked to the cluster with the cluster number 11. Similarly, "12" and "13" are stored as the FAT entry corresponding to the cluster number 12 and the FAT entry corresponding to the cluster number 13, respectively and the clusters are linked to each other in sequence such as cluster numbers 10, 11, 12 and 13. Next, "0xFFF" is stored as the FAT entry corresponding to the cluster number 13. Since "0xFFF" means termination of the link, the link starting from the cluster number 10 terminates in four clusters 10, 11, 12 and 13. "0" stored as the FAT entry corresponding to the cluster number 14 indicates that the cluster is not assigned for the file and serves as a free area.

When it is recognized that the data area assigned for the file FILE1. TXT is the clusters with the cluster numbers 10, 11, 12 and 13 as shown in Figs. ~~2(A) and 2(B)~~ 2A and 2B, in reading

the data in the file FILE1. TXT, data of the cluster numbers 10, 11, 12 and 13 in the data area 203 is read sequentially as shown in Fig. ~~2(C)~~ 2C.

Referring to Fig. 3, an example of writing the file data in the FAT file system will be described below. As in the example in Fig. 2, it is assumed that the directory entry 201 shown in Fig. ~~3(A)~~ 3A is stored in a part of the route directory entry or data area. In a file indicated in the directory entry 201, the file is named as FILE1. TXT and stores its data starting from the cluster number 10. Since the file size is 16000 bytes and size of one cluster is 4096 in the example in Fig. 3, the file data is stored in four clusters.

Fig. ~~3(B)~~ 3B shows the state of the FAT 202 prior to writing to the file. As in the case in Fig. ~~2(B)~~ 2B, the FAT in Fig. ~~3(B)~~ 3B shows that four clusters 10, 11, 12 and 13 are linked with, and that the data of the file FILE1. TXT is stored in four clusters 10 to 13.

Please substitute the paragraph beginning on page 11, line 23 and ending at page 12, line 16 to read as follows:

Assignment of an free area is achieved by acquiring the free cluster from the FAT and changing the link of the FAT. A link changing procedure is as follows. First, an entry storing "0" representing a free cluster is acquired from the FAT 202 in Fig. ~~3(B)~~ 3B. In the case of Fig. ~~3(B)~~ 3B, a cluster with the cluster number 14 is a free cluster. Subsequently, the acquired free cluster is linked to termination of the link of the file to be extended its file size. Fig. ~~3(C)~~ 3C shows the state of the FAT 202 after changing the link and a destination to be linked to the cluster number 13 as the end of FILE1. TXT is changed to "14". A link destination to the cluster number 14 is changed to "0xFFF" indicating the end of link. According to this processing, as shown in Fig. ~~3(D)~~ 3D, in FILE1. TXT, five clusters of "10", "11", "12", "13" and "14" are assigned as data region 203A for the file, 16001^{st} to 16384^{th} bytes are written to the area with the cluster number 13 and 16385^{th} to 17000^{th} bytes are written to the area with the cluster number 14 to perform writing of data.

Please substitute the paragraphs beginning on page 16, line 24 and ending at page 17, line 14 to read as follows:

In the example in Fig. ~~5(A)~~ 5A, four cache blocks exist in the FAT cache 501 and the FAT is read from the information recording medium 110. The block size of the cache block is 32 sectors as the erase block size or one sector as the minimum access size. In the example in Fig. ~~5(A)~~ 5A, the block size of the cache blocks 1 and 4 is 32 sectors and the block size of the cache blocks 2 and 3 is 1 sector. Remaining area is managed as the free area and used as the area in which a cache block is newly generated in case of mishit in cache.

Fig. ~~5(B)~~ 5B is a view showing an example of FAT cache management information 502 which stores management information of the FAT cache 501 therein. Fig. ~~5(B)~~ 5B corresponds to the FAT cache 501 in Fig. ~~5(A)~~ 5A. The FAT cache management information 502 includes a block start address, FAT address, FAT size and update flag.

Please substitute the paragraph beginning on page 17, line 22 and ending at page 18, line 12 to read as follows:

In the example of Fig. ~~5(B)~~ 5B, the cache block 1 exists from a leading position of the FAT cache 501, the FAT for 22 sectors is read from the head of the FAT of the information

recording medium and a part of the FAT in the cache block 1 is updated. In this embodiment, the leading position is represented as "1", not "0". The cache block 2 exists from 33rd sector of the FAT cache 501, the FAT for 1 sector is read from the 60th sector of the FAT in the information recording medium and the FAT in the cache block 2 is not updated. Furthermore, "0xFFFF" as information on address and size is set in the FAT cache management information 502 corresponding to the cache block 5, which indicates that there exists no cache block 5. Here, a decimal value 65535 corresponding to "0xFFFF" is a value which is not used as a valid address or size in this embodiment.

Please substitute the paragraph beginning on page 26, line 17 and ending at page 27, line 5 to read as follows:

As shown in Fig. ~~5(B)~~ 5B, in the embodiment of the present invention, an example in which a set of four pieces of information consisting of the block start address, FAT address, FAT size and update flag is managed as the FAT cache management information 502 is described. However, as long as the FAT cache can be managed using similar information, the other form is available. Furthermore, in the free area retrieval processing, the following example is described: the cluster number at a last

finishing location of the free area retrieval is held at a starting cluster number SCN of the free area retrieval and is used in the next free area retrieval processing. However, random number or the other value obtained, for example, by setting the head of the FAT each time, may be used.

Please substitute the paragraph beginning on page 29, line 3 and ending at line 17 to read as follows:

Fig. ~~9(B)~~ 9B is an explanation view showing an example of a FAT cache management information 903 which stores information for managing the FAT_Read cache 901 and FAT_Write cache 902. Fig. ~~9(B)~~ 9B corresponds to the FAT_Read cache 901 and FAT_Write cache 902 in the state in Fig. ~~9(A)~~ 9A. The FAT cache management information 903 includes FAT address, FAT size and update flag. The FAT address indicates a region of a FAT read into the cache block lies in the FAT. The FAT size indicates the size of the FAT read into the cache block. The update flag is a flag which indicates whether or not the FAT in the cache block is updated. Using the three pieces of information as a set, the FAT cache management information 903 includes M sets of information for

FAT_Read cache 901 and N sets of information for FAT_Write cache 902.